Редактор Миссий

Формат миссии .mit:	Миссия состоит из двух файлов: имя_миссии.lua и имя_миссии.mit.

Формат mit файла:

Title	Идентификатор строки из базы данных для названия компании (используется поле Title)
Info	Идентификатор строки из базы данных для описания компании (используется поле Text)
Map	Имя файла карты.
MyPlayer	Номер игрока.
PlayersQuantity	Общее число игроков на карте.

Далее для каждого игрока:

AIType	Число. Тип АИ. См. описания в разделе «Заметки»
Team	Номер команды (начинается с 0)
Gold	Кол-во золота
Iron	Кол-во железа
Wood	Кол-во дерева
Food	Кол-во еды
Oil	Кол-во нефти.

Формат компании .cmt: Любой файл с расширением .cmt в папке \Data\Campaigns\ рассматривается движком как компания.

Формат файла:
Идентификатор строки из базы данных для названия компании (используется поле Title)
Идентификатор строки из базы данных для описания компании (используется поле Text)
Название портрета (vilgelm_, angl_, franz_, jofr_).
Имя_файла_миссии_без_расширения_1
Имя_файла_миссии_без_расширения_2
....
Имя_файла_миссии_без_расширения_n

Локализация: Файлы с локализационной информацией:
\Data*.tt
\Data*.st – эти файлы менять не надо.
Файлы \Data*.tt получаются экспортом из базы данных custom.mdb. В ней содержится таблица, 1 пример текста в ней, и макрос экспорта.

Список функций:

void SetShowFogForAllies(bool aIsShow)
— Включает/выключает туман войны для союзников. Данный атрибут НЕ СОХРАНЯЕТСЯ в сейв!! Поэтому вызов этой функции должен находиться в глобальном пространстве, а не в Init().

void MessageBox(string StringID)
— Выводит сообщение. Если StringID отсутствует в локализационной базе, то выводится он сам.

float GetTime(void)
— Возвращает игровое время (которое, например, на паузе не идёт) в секундах.

void SetPlayerAIType(int aPlayer, int AITemplateID)
— Устанавливает для игрока aPlayer тип интеллекта:

AITemplateID = AITypePassiveConst	= 0	— Пассивный АИ.
AITemplateID = AITypeDefendCons	= 1	— Защитный АИ.
AITemplateID = AITypeDefendAttackConst	= 2	— Смешанный АИ.
AITemplateID = AITypeAttackConst	= 3	— Агрессивный АИ.
AITemplateID = AITypeAirAttackConst	= 4	
AITemplateID = AITypeSeaAttackConst	= 5	

bool IsPlayerUnitInArea(int aLeft, int aTop, int aRight, int aBottom, int aPlayer)
— Возвращает true, если в указанном прямоугольнике находится хотя бы один живой юнит игрока aPlayer.

void OpenMapToPlayer(int aX, int aY, int aR)
— Открывает игроку кусок карты.

void SetCamera(int aX, int aY)
— Перемещает камеру в указанную точку карты.

bool IsPlayerHouseInArea(int aLeft, int aTop, int aRight, int aBottom, int aPlayer, int aType)
— Возвращает true, если в указанном прямоугольнике находится хотя бы одно зданите игрок — aPlayer и типа — aType.

Типы:

HOUSE_NEUTRAL_BRIDGE1	= 0
HOUSE_NEUTRAL_BRIDGE2	= 1
HOUSE_NEUTRAL_BRIDGE_LAST	= 1
HOUSE_POWERPLANT	= 0
HOUSE_DOT4	= 1
HOUSE_ACADNAUK	= 2
HOUSE_ACADSTROIT	= 3
HOUSE_KAZARMA	= 4
HOUSE_ACADWAR	= 5
HOUSE_TOWN_CENTER	= 6
HOUSE_DOT6	= 7
HOUSE_MARKET_PLACE	= 8
HOUSE_SMALL_HOUSE	= 9
HOUSE_IRON_MAKER	= 10
HOUSE_GOLD_MAKER	= 11
HOUSE_WOOD_MAKER	= 12
HOUSE_OIL_MAKER	= 13
HOUSE_MINT	= 14
HOUSE_AMBAR	= 15
HOUSE_CANNON_DEPOT	= 16
HOUSE_HOSPITAL	= 17
HOUSE_MEDIC_SPOT	= 18
HOUSE_TANK_DEPOT	= 19
HOUSE_AVIA_HANGAR	= 20
HOUSE_WATCH_TOWER	= 21
HOUSE_PORT	= 22
HOUSE_STABLE	= 23
HOUSE_GAUBICA_ST	= 24
HOUSE_ALL	= -1

int GetPlayerStatsKilled(int aPlayer)
— Возвращает число убитых юнитов у игрока aPlayer.

int GetPlayerStatsUnits(int aPlayer)
— Возвращает число живых юнитов у игрока aPlayer.

int GetPlayerStatsHouses(int aPlayer)
— Возвращает число домов у игрока aPlayer.

void LuaManager_StopCurrentScript(int Winner)
— Останавливает скрипт, «назначает» победителем игрока — Winner, выходит из миссии.
void UnitsGroup_Create(string Name)
— Создаёт пустую группу.

void UnitsGroup_Remove(void)
— Удаляет группу (но не сами юниты).

void UnitsGroup_AddInRect(string aGroupName, LuaObject aRect, int aPlayer, int aType, int aMaxUnitsQuantity)
— aRect — таблица с полями : left, top, right, bottom
— aMaxUnitsQuantity — максимальное число юнитов.
— Функцию можно вызвать только для пустой группы.

void UnitsGroup_SendTo(string aGroupName, int aTargetX, int aTargetY, LuaObject aParams)
void UnitsGroup_SendToByType(string aGroupName, int aTargetX, int aTargetY, LuaObject aParams)
void UnitsGroup_SendToMulti(string aGroupName, int aTargetX, int aTargetY, LuaObject aParams)
— Послать все юниты в группе в путь.
— Если координаты являются координатами дома, а сказано, что точки — поведение не определенно.
— ByType — группирует юниты по типам и посылает.
— Multi — посылает в виде множества групп.
— aParams — таблица с полями : AiLevel, TemplateType, TemplateCount, TargetType, TargetPlayer, TargetNum, MayDismiss, NoGroup.
— Описание полей см. в «Заметки».

bool UnitsGroup_IsInRect(string aGroupName, LuaObject aRect)
— Возвращает true если хотя бы один юнит входящий в группу aGroupName, находится в прямоугольнике — aRect.

int UnitsGroup_GetUnitsQuantity(string aGroupName)
— Возвращает число юнитов в группе.

void Timer_Create(string aTimerName, string aStringToDo, float aDelay, uint aMaxCount)
— Выполняет aMaxCount раз, строку скрипта aStringToDo, каждые aDelay секунд.
— Имя каждого таймера обязано быть уникальным.
— Вечного таймера нет, но aMaxCount можно сделать очень большим.
— Таймер, который закончил свою работу (сработал aMaxCount раз), автоматически удаляется.

void Timer_Destroy(string aTimerName)
— Удаляет таймер

int GetHousesQuantity(int aPlayer, int aType)
— Возвращает число домов типа aType у игрока aPlayer.

void AddMessage(string Prefix, int Value, string Postfix)
— Формирует строку : Prefix + Value + Postfix
— Отображает её в верхнем левом углу экрана.

Заметки:

— Везде, где задаются координаты, имеет смысл использовать Red_comp для их определения.
— Параметры по умолчанию для «посыла» группы имеет смысл объявить один раз и больше не трогать

```
DefaultParams = {}
DefaultParams.AiLevel          = GROUP_AI_LEVEL_AUTONOM
DefaultParams.TemplateType     = TEMPLATE_TYPE_RECT
DefaultParams.TemplateCount    = 0
DefaultParams.TargetType       = TARGET_TYPE_PLACE
DefaultParams.TargetPlayer     = 1
DefaultParams.TargetNum        = 0
DefaultParams.MayDismiss       = true
DefaultParams.NoGroup          = true
```

TemplateType — тип шаблона построения

TEMPLATE_TYPE_LINES = 1 — колонны в несколько колонн
TEMPLATE_TYPE_RECT = 2 — квадратом сплошным
TEMPLATE_TYPE_CEPOCHKA = 3 — цепочкой — атака на цель цепью в несколько рядов

TemplateCount — число для шаблона — если 0, то подбирается автоматически

TargetType — Тип цели, куда идут

TARGET_TYPE_NONE = 0— просто сформировать группу, которая будет стоять на месте
TARGET_TYPE_PLACE = 1— идем в точку
TARGET_TYPE_ALIEN_HOUSE = 2 — атакуем чужой дом
TARGET_TYPE_UNIT = 3 — идем атаковать юнита
TARGET_TYPE_MISC = 4 — идем в кукурузу
TARGET_TYPE_TRENCH = 5 — идем в окоп
TARGET_TYPE_HOUSE = 6 — в свой дом послали

```
TargetCoord  — Координаты куда идут в пикселях
_AiLevel  — уровень командира группы:
GROUP_AI_LEVEL_NONE       = 0 — нет командира
GROUP_AI_LEVEL_1          = 1 — следит за бомбами
GROUP_AI_LEVEL_2          = 2 — нет командира
GROUP_AI_LEVEL_3          = 3 — нет командира
GROUP_AI_LEVEL_AUTONOM    = 4 — командир идет
                                по своему пути
```

если понадобится изменить поведение по умолчанию, то удобно использовать конструкцию:

```
tParam = DefaultParams
tParam.NoGroup = false

UnitsGroup_SendTo(«Attackers», 132, 176, DefaultParams) — по умолчанию
UnitsGroup_SendTo(«Attackers», 132, 176, tParams) — всё тоже, что по умолчанию, по идём строем.
```

При использовании таймеров, нужно помнить, что, в выполняемой строке нельзя ни в коем случае использовать локальные переменные. Это приведёт к проблемам с сохранением/загрузкой.

— Все переменные объявленные в глобальном пространстве имён и/или без ключевого слова «local» сохраняются в save игры.
— Весь код вне тела какой-либо функций выполняется при каждой загрузке скрипта (и при загрузке сохранения и при старте миссии). Поэтому, для корректной загрузки из save нужно использовать следующую конструкцию:

```
MustBeInitialized = true

function Init()
        ... обявления всех глобальных переменных и инициализация их значениями для _новой_ игры ...
end

function main()

        if MustBeInitialized == true then
                Init()
                MustBeInitialized = false
        end
        ... Весь остальной код ....
end
```

— Каждый кадр вызывается функция — main()